High-performance I/O on Cray T3E

Ulrich Detert
U.Detert@fz-juelich.de
Institute for Applied Mathematics
Research Center Jülich, Germany

A great number of hardware and software features allow for potentially high I/O performance on Cray T3E systems. In the following, the influence of RAID technology, file striping, global I/O FFIO layers, and pcache on I/O performance are discussed. The I/O performance achieved with striped SCSI disks is compared to RAID disk performance and tuning opportunities on the system and the user level are evaluated..

Introduction

Cray T3E systems provide for a potentially very high computational performance. For a balanced system architecture, however, not only the computational performance, but also the I/O performance is of great importance. Large technical and scientific applications tend to process enormous amounts of data and are often structured in a way such that very long-running jobs are separated into segments which are processed one after the other. Each segment typically runs for only several minutes to few hours, reads the results of the previous segment and produces intermediate results written onto disk for use by the next segment. In order to be able to compete with the computational speed the I/O implementation has to use parallelism in a similar manor like the computation itself. RAID technology, file striping and global I/O are of great importance in this respect. Furthermore caching of data can be very profitable, especially, if a large number of small I/O requests has to be processed. Cray T3E systems provide pcache on the device level for these ends.

Unicos/mk 2.0 allows for check-pointing of jobs, e.g. to avoid production losses during maintenance periods. Since jobs on a big T3E system can potentially use very large amounts of memory - up to about 60 GB on the T3E-512 system in Jülich – good I/O performance is required for the check-pointing of jobs as well.

In the following, basic properties of the I/O configuration of the Jülich T3E-512 system are described. Then, the I/O performance achieved using various tuning alternatives is discussed.

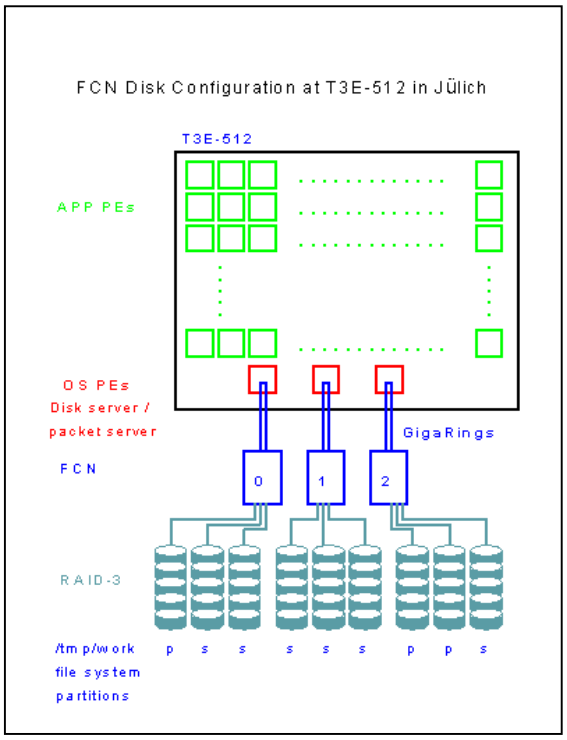

Fig. 1: T3E I/O Configuration

T3E I/O Configuration in Jülich

The current T3E I/O configuration in Jülich comprises 9 FCN RAID disks attached to 3 FCN controllers (Fig. 1). The largest file system – and the one used for

the following performance tests – is /tmp/work with a capicity of ca. 260 GB. /tmp/work is striped on 9 RAIDS using 3 primary and 6 secondary partitions. Only the secondary partitions are relevant for performance, if user striping and the global I/O FFIO layer are used.

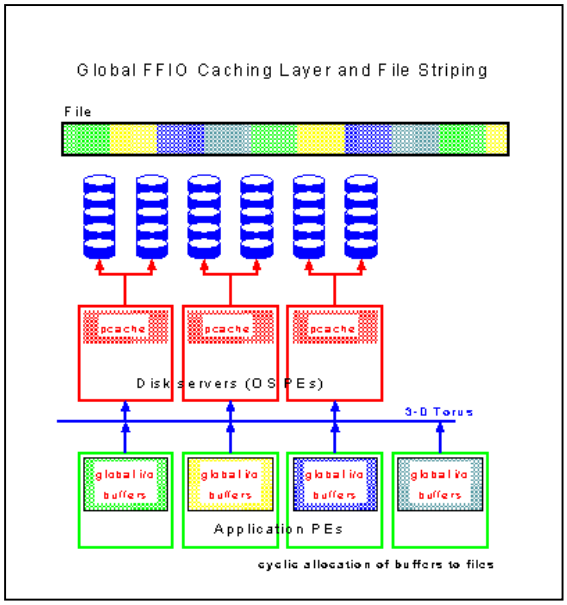

Fig. 2: Global I/O Caching Layer

The Global I/O FFIO Layer

For the performance measurements presented in the following, global I/O and user striping of data have been used. Both can be switched on with the assign command. The *–p* option defines the (secondary) partitions to be used for striping, *-q* defines the block size for each stripe, *-F global* switches on global I/O, and *size* and *no* define the size and number of cache pages to be used for the global I/O caching layer on each application PE. Optimal performance can be expected, if the global I/O cache page size is a multiple of the partition block size defined via *–q* and the partition block size on the other hand is a multiple of the file system´s allocation unit size (64 KB for the considered file system). Furthermore, the user application should use I/O request sizes that fit the global I/O cache page size. If all above conditions are met, we talk about **wellformed** I/O.

Fig. 2 depicts the principles of operation of global I/O. One important

```
assign
  -p part_list
  -q part_size
  -F global:size:no
```

feature of global I/O is that the global I/O cache is distributed among all application PEs used by the application program. The mapping between cache pages and file positions is implemented in a cyclic manor. As a consequence, if an application program follows this cyclic allocation scheme, only data cached in the local cache pages are accessed. If, on the other hand, the application program does not follow this scheme, data has first to be sent to the cache page that corresponds to the addressed file position and will then be sent to disk.

Disk Servers, Packet Servers, and pcache

Disk servers and packet servers are located on OS PEs. Packet servers handle I/O packages sent to or received from GigaRings connected with FCN nodes. There is one packet server per GigaRing. Disk servers handle the transport of I/O requests to or from disks. Each disk server can control one or more disks. Physical device cache (pcache) can be allocated on each disk server for individual physical slices.

For the performance measurements presented here, two variants have been tested:

- three OS PEs, each hosting one disk server and one packet server (each disk server controlling three disks)
- six OS PEs, three of them hosting one disk server and one packet server each, and the other three hosting one disk server each (each disk server controlling two disks)

The potential advantage of using multiple disk servers is the ability to

allocate more memory for pcache. This is especially important, if there is only a small amount of memory available on each PE (128 MB per PE at the T3E system in Jülich).

Results

For the proper assessment of tuning techniques it is important to know, how reproducible measurement results are. Fig. 3 depicts the variation of the results obtained in two identical tests on a dedicated T3E-512 system.

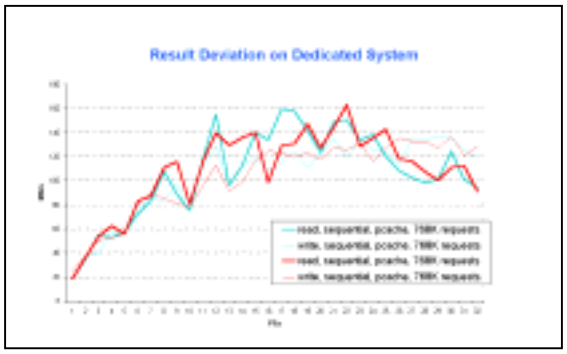

Fig. 3: Result Deviation on Dedicated System

The effect of wellformed I/O vs. malformed I/O is shown in Fig. 4. I/O requests of size 800 000 bytes neither match the size of global I/O cache pages (defined in terms of 4-K blocks) nor the disk allocation unit size of 64 KB.

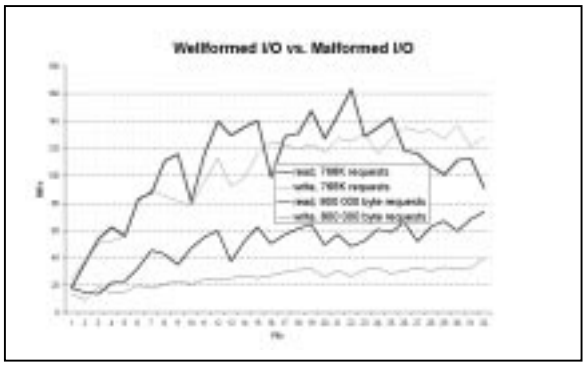

Fig. 4: Wellformed I/O

Fig. 5 depicts the performance gain achieved by adjusting the application program´s I/O scheme to the cyclic mapping of global I/O cache pages to file blocks.

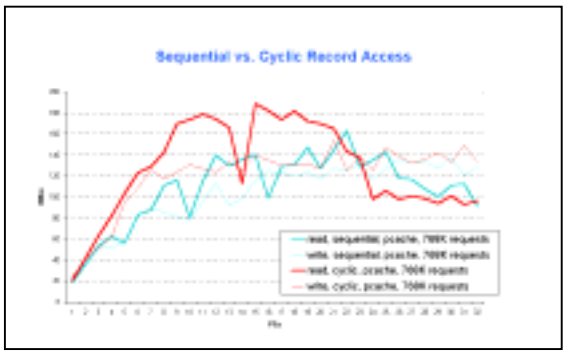

Fig. 5: Cyclic Record Access

Compared to the relatively large amount data moved in the I/O requests used for these performance tests (ca. 8 MB per PE), the amount of memory available for pcache is rather small (ca. 18 MB per disk server). Hence, there is practically no data re-use from pcache. As a consequence, only write operations can profit from pcache. Reads are slowed down, instead – especially for large numbers of PEs (Fig. 6).

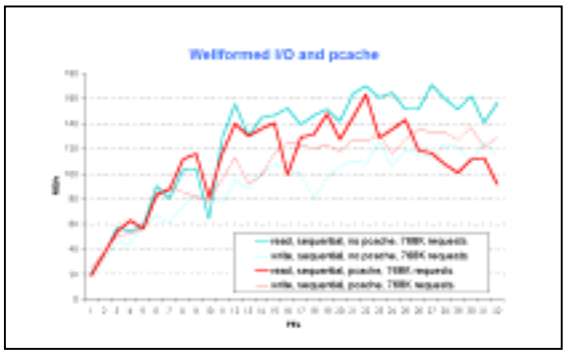

Fig. 6: Wellformed I/O and pcache

Fig. 7 and Fig. 8 exhibit the effect of defining additional OS PEs as disk servers. Without the allocation of additional pcache (Fig. 7) there is practically no effect on performance.

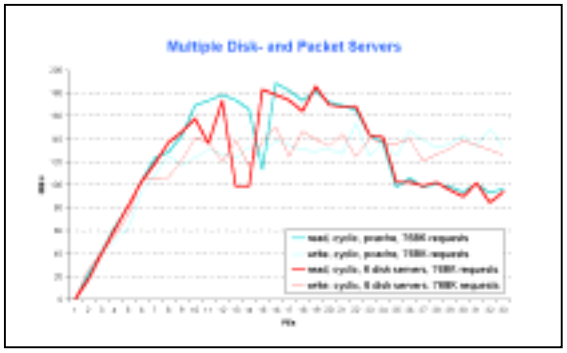

Fig. 7: Multiple Disk Servers

If, on the other hand, additional memory is assigned for pcache, there may be a significant performance improvement, since data re-use is more likely to occur (Fig. 8).

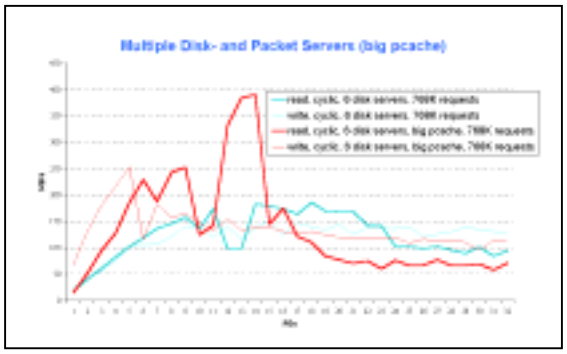

Fig. 8: Multiple Disk Servers and pcache

SCSI vs. FCN RAID Disks

RAID-3 disks provide for better data safety as compared to SCSI disks due to the inherent data redundancy. Besides, better performance should be expected, since parallel access to a cluster of disks is provided. Fig. 9 compares the performance obtained with the RAID configuration described in the previous sections with those achieved with 32 independent SCSI disks, used in parallel by means of global I/O and file striping.

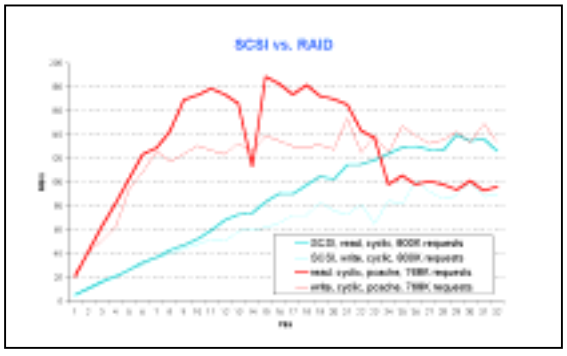

Fig. 9: SCSI vs. FCN RAID Disks

Summary

As a result of the above performance measurements we state that tuning options available to the application programmer are in general much more important for performance than modifications related to the operating system level. Above all, wellformed I/O is a prerequisite for good I/O performance. Proper calibration of the I/O request size and the size and number of global I/O cache pages should be aimed at. An additional performance gain can be achieved by following the cyclic mapping of global I/O cache pages to file blocks.

The allocation of pcache on disk servers was not of significant importance in our measurements. This is mostly due to the fact that only small amounts of memory could be made available for pcache. Hence, large I/O requests like those considered here could not profit from pcache.

The assignment of additional disk servers to multiple OS PEs did not improve performance significantly, unless the additional disk servers were equipped with pcache. This proves that disk servers controlling three RAID disks each - as configured at Research Centre Jülich - are not a bottleneck for performance.

The comparison of SCSI disks and FCN RAID disks shows that RAID disks are significantly superior to SCSI disks with respect to performance.